An Efficient Local Approach to Convexity Testing of Piecewise-Linear Hypersurfaces

Konstantin Rybnikov

*One University Ave., Olney Hall 428, University of Massachusetts at Lowell,
Lowell, MA 01854 USA*

Email address: krybniko@cs.uml.edu

Abstract

We show that a closed piecewise-linear hypersurface immersed in $\mathbb{R}^n$ ($n \geq 3$) is the boundary of a convex body if and only if every point in the interior of each $(n-3)$-face has a neighborhood that lies on the boundary of some convex body; no assumptions about the hypersurface's topology are needed. We derive this criterion from our generalization of Van Heijenoort's (1952) theorem on locally convex hypersurfaces in $\mathbb{R}^n$ to spherical spaces. We also give an easy-to-implement convexity testing algorithm, which is based on our criterion. For $\mathbb{R}^3$ the number of arithmetic operations used by the algorithm is at most linear in the number of vertices, while in general it is at most linear in the number of incidences between the $(n-2)$-faces and $(n-3)$-faces. When the dimension n is not fixed and only ring arithmetic is allowed, the algorithm still remains polynomial. Our method works in more general situations than the convexity verification algorithms developed by Mehlhorn et al. (1996) and Devillers et al. (1998) – for example, our method does not require the input surface to be orientable, nor it requires the input data to include normal vectors to the facets that are oriented "in a coherent way". For $\mathbb{R}^3$ the complexity of our algorithm is the same as that of previous algorithms; for higher dimensions there seems to be no clear winner, but our approach is the only one that easily handles inputs in which the facet normals are not known to be coherently oriented or are not given at all. Furthermore, our method can be extended to piecewise-polynomial surfaces of small degree.

Key words: Program checking, Output verification, Geometric Property Testing, Convexity, Piecewise-linear surface

Time 00:28   1 February 2008

1 Introduction

Blum and Kannan (1989) suggested a paradigm of output verification. Since a complete check of a program is often difficult or not possible – for example, when the source code has not been made public – it is important to have algorithms that verify key properties of mathematical objects generated by programs. Instead of the source code verification one can try to verify the properties of the output that are deemed essential by users of the program. In computational geometry this paradigm was developed, among others, by Melhorn et al. (1996, 1999) and Devillers et al. (1998). For example, the LEDA C++ library contains programs verifying convexity of a polygon, Delaunay property of a tiling, etc (Mehlhorn and Näher, 2000). Devillers et al. (1998) argue that it is easier to evaluate the quality of the output of a geometric algorithm, than the correctness of the algorithm or program producing it. This paper contributes to the problem of verification of convexity of a large class of piecewise-linear (PL) hypersurfaces in $\mathbb{R}^n$ for $n \geq 3$. The novelty of our approach is in reducing the verification of global convexity of a PL-hypersurface to the verification of local convexity at the faces of small codimension. We show that a closed bounded PL-hypersurface realized in $\mathbb{R}^n$ ($n \geq 3$) without local self-intersections is the boundary of a convex body if and only if each $(n-3)$-face of the hypersurface has a point in its relative interior such that a small Euclidean ball at this point is cut by the hypersurface into two pieces, one of which is convex. The local convexity condition can also be expressed as that the point has a neighborhood on the hypersurface, which lies on the boundary of a convex body. If the hypersurface is not bounded, this criterion is invalid. However, *if* it is known that the hypersurface has *at least one point of strict convexity* – i.e., a point s such that a small ball, centered at s, intersects the hypersurface over a set S, which lies on the boundary of a convex body and such that $S \setminus s$ lies in an *open halfspace* with respect to some hyperplane through s – our test can still be used. In fact, our result is slightly more general: we prove our criterion for any cellwise-flat PL-realization in $\mathbb{R}^n$ ($n \geq 3$) of a semiregular CW-partition of an $(n-1)$-manifold. The technical terms used in this formulation are defined in the next section.

In this paper we also construct an algorithm for convexity testing that can be applied to any closed PL-hypersurface. Our approach does not require any preliminary knowledge of the topology of the surface, or any information about its orientability. The direct comparison of the complexity of our algorithm and those of Mehlhorn et al. (1996b; 1999) and Devillers et al. (1998) is not quite meaningful, since these authors make the following simplifying assumptions.

(CO) *The input is known to be an orientable closed hypersurface. The normals to the $(n-1)$-faces are given as part of the input, and they are all oriented either outwards or inwards ("coherent orientation")*

(S) *the cell-partition is simplicial.*

The last condition is not needed for these algorithms to work; however, it does affect the complexity analysis, which is given in the papers only for the simplicial case. It is not clear to us why the assumption (CO) is natural; in any case, they significantly simplify all considerations. Our approach does not require the assumption (CO). For $\mathbb{R}^3$, if conditions

(CO) and (C) are met, both our and previous approaches have the same complexity, which is big-O in the number of vertices. While Devillers et al.(1998) claimed $O(f_0)$ as the running time of for any dimension, this was clearly a typo: first, this bound is impossible, even for PL-spheres, by fundamental counting theorems of polyhedral combinatorics and second, the pseudocode in (Devillers et al, 1998: Section 3) has running time, in the notation of that paper, of $\sum\limits_{j=0}^{d-2} \mathsf{f}_j + \sum\limits_{\{F\in\Gamma \mid \dim F=d-2\}} \sum\limits_{j=3}^{d} \mathsf{f}_{j-1,j}(F)$.

Formal definitions and notation are given in Section 2. The paper presupposes some familiarity with partially ordered sets (posets), linear algebra and geometry (e.g. Kostrikin and Manin, 1988), basic convexity theory (e.g. Rockafellar, 1990), general topology (e.g. Dugundji, 1965), and basic combinatorial topology (e.g. Seifert and Threlfall, 1980).

1.1 On efficiency of Convexity Checkers

Suppose all computations are done with floating point arithmetic. More formally, consider the random access machine (RAM) with unit cost model of computation, where all four arithemtic operations are included in the instruction set. As usual, $\mathsf{f}_{i\,j}$ denotes the number of incidences between i-dimensional and j-dimensional faces and $\mathsf{f}_i = \mathsf{f}_{i\,i}$ stands for the number of i-dimensional faces. Consider the case of $\mathbb{R}^3$. If condition (CO) holds, the algorithms by Mehlhorn et al. (1996) and Devillers et al. (1998), have the same time-complexity of $O(\mathsf{f}_0)$. Our algorithm also has the complexity of $O(\mathsf{f}_0)$, without requiring the assumption (CO); furthermore, its working does not depend on whether the surface is orientable or not. In a more general situation, where the conditions (CO) and (S), or some of them, cannot be assumed, the complexity of our algorithm, as well as the algorithms of the previous authors heavily depend on the following three factors: i) the combinatorics of the cell-partition (e.g. simplicial or not), ii) the geometry of the realization (e.g. generic positions of the vertices vs. completely general case), iii) the form of the input.

Regarding iii), for example, the combinatorial information about the input can be given in the form of the complete poset of faces, or some subposet of faces, such as, e.g. the vertex-facet graph. Furthermore, certain additional topological information (e.g. the knowledge that the hypersurface is orientable, or a cyclic order of the facets at each $(n-3)$-face) might speed up the convexity verification. The geometry of the realization can be given by the equations of the facets, or by "coherent" inequalities for the facets (CO), or by positions of the vertices, or in the form of inner normals at $(n-3)$-cells to the $(n-2)$- and $(n-3)$-cells.

Now let us consider the general problem of convexity verification. At one end is the simplified setup, where the input hypersurface is simplicial and the realization is sufficiently generic so that floating point arithmetic can safely be used. Under these assumptions everything is fast, regardless of the method used. On the other end is the completely general setup, where nothing can be assumed. One can also consider "intermediate" models, such as, e.g., where the input hypersurface is simplicial, but the positions of the vertices are not necessarily generic. Another reasonable assumption would be that the hypersurface is not necessarily simplicial, but the realizations of the facets are known to be convex. We give our algorithm for the most general case, where nothing can be assumed. One of the motivations for this generality is the work of Joswig and Ziegler (2004), who clearly demonstrated that from the complexity theory point of view the convex hull problem is most interesting when we cannot assume that the vertices (or hyperplanes) are in general position – or that the dimension of the space is a small fixed number.

If sufficient linear-algebraic data and face incidence information are given about the stars of $(n-3)$-faces (see Section 6.3 for details), then the complexity of our algorithmic approach is still polynomial (in the Turing machine model). The distinguishing features of our approach are: 1) locality of testing and 2) independence from the global topology of the input surface. Moreover, our approach generalizes to piecewise-algebraic surfaces (this work is in progress).

2 Definitions and Background

From now on $\mathbb{X}^n$ is used to denote $\mathbb{R}^n$ or $\mathbb{S}^n$. By a *subspace* of $\mathbb{X}^n$ we mean an affine subspace in the case of $\mathbb{R}^n$, and the intersection of $\mathbb{S}^n \subset \mathbb{R}^{n+1}$ with a linear subspace of $\mathbb{R}^{n+1}$ in the case of $\mathbb{S}^n$. A *hyperplane* is a subspace of codimension one. A set $K \subset \mathbb{X}^n$ is called *convex* if for any $x, y \in K$ there is a minimal geodesic segment with end-points x and y that lies in K. Then, the dimension of K, $\dim K$, is understood as the dimension of a minimal subspace containing K (since such a subspace is unique for both $\mathbb{R}^n$ and $\mathbb{S}^n$, we denote it by $\operatorname{aff} K$, even in the spherical case). We use $\operatorname{int} K$ to denote the interior of K in $\operatorname{aff} K$: in other words, $\operatorname{int} K$ stands for the *interior* of K *relative* to $\operatorname{aff} K$. A *hypersurface in* $\mathbb{X}^n$ is a pair $(\mathscr{M}, r)$ where $\mathscr{M}$ is a manifold of dimension $n-1$, with or without boundary, and $r : \mathscr{M} \to \mathbb{X}^n$ is a continuous *realization* map. Unless specified otherwise, a *manifold* will always mean a manifold without boundary. When the realization map is fixed we do not mention it every time. A realization r is called complete if any sequence on $\mathscr{M}$, which is Cauchy with respect to the r-metric on $\mathscr{M}$ (see Section 3), converges to a point of $\mathscr{M}$. A map $i : \mathscr{M} \to \mathbb{X}^n$ is called an *immersion* if i is a local homeomorphism, in such a case we may also refer to $(\mathscr{M}, i)$ as an immersed hypersurface. A map $f : \mathscr{M} \to \mathbb{X}^n$ is called an *embedding* if f is a homeomorphism onto $f(\mathscr{M})$. Obviously, an embedding is an immersion, but not vice versa. We call a k-submanifold (with or without boundary) $\mathbb{S}$ of $\mathscr{M}$ flat with respect to the realization map r if $r : \mathbb{S} \to r(\mathbb{S})$ is an embedding into a k-subspace of $\mathbb{X}^n$. A submanifold is called open if it is non-compact and without boundary.

A *convex body* in $\mathbb{X}^n$ is a closed convex set of full dimension. The hypersurface $(\mathscr{M}, r)$ is called *locally convex at* $p \in \mathscr{M}$ if p has a neighborhood $\mathscr{N}_p \subset \mathscr{M}$ such that $r : \mathscr{N}_p \to r(\mathscr{N}_p)$ is a homeomorphism and $r(\mathscr{N}_p)$ lies on the boundary of a convex body $K_p = K_p(\mathscr{N}_p)$. Often, when it is clear from the context that we are discussing the properties of r near $\mathbf{p} = r(p)$, we say that r is convex at $\mathbf{p}$. If K_p can be chosen so that $K_p \setminus r(p)$ lies in an open half-space defined by some hyperplane passing through $r(p)$, the realization r is called *strictly convex* at p. We will also sometimes refer to $(\mathscr{M}, r)$ as strictly convex at $r(p)$. By a theorem of Busemann (1958) $\mathscr{N}_p$ and $K_p(\mathscr{N}_p)$ can always be chosen to allow a support hyperplane H at $r(p)$ such that the orthogonal projection of $r(\mathscr{N}_p)$ onto H is an open $(n-1)$-ball. Due to the local nature of this theorem, it holds in all spaces of constant curvature. When K_p and $\mathscr{N}_p$ satisfy this assumption, we refer to K_p as a *convex witness* for p.

Let us recall (e.g. Rockafellar, 1990) that a point p on the boundary of a convex set K is called *exposed* if K has a support hyperplane that intersects $\overline{K}$, the closure of K, only at p; also, p is called *extreme* if it does not belong to the interior of any interval contained in ∂K. Thus, an *exposed* point on a convex body B is a *point of strict convexity* on the hypersurface ∂B. Conversly, a point of strict convexity $p \in \mathscr{M}$ for $(\mathscr{M}, r)$ is an exposed point for a convex witness K_p. Local convexity can be defined in many other, non-equivalent, ways (e.g., see van Heijenoort, 1952).

We will say that the hypersurface $(\mathscr{M}, r)$ is *the boundary of a convex body* $K \subset \mathbb{X}^n$ if r is a homeomorphism from $\mathscr{M}$ onto ∂K. Hence, we exclude the cases when $r(\mathscr{M})$ coincides with the boundary of a convex body but r is not injective. Our algorithm for PL-hypersurfaces will always detect a violation of the immersion property; in the case where $r(\mathscr{M})$ is the boundary of a convex body, but r is not a homeomorphism, it will produce the negative answer without trying to determine if $r(\mathscr{M})$ is the boundary of a convex body. Of course, the algorithmic and topological aspects of this case may be interesting to certain areas of geometry, such as origami. Note that for $n \geq 3$ a closed $(n-1)$-manifold $\mathscr{M}$ cannot be immersed into $\mathbb{R}^n$ by a non-injective map r so that $r(\mathscr{M})$ is the boundary of a convex set – any convex hypersurface in $\mathbb{R}^n$ is simply-connected and any covering map onto a simply-connected manifold must be a homeomorphism. However, such immersions cannot be easily ruled out in the hyperbolic space $\mathbb{H}^n$, as there are infinitely many topological types of convex hypersurfaces in $\mathbb{H}^n$ for $n > 2$ (Kuzminykh, 2005).

This paper is mainly focused on convexity of piecewise-linear (PL) hypersurfaces, in particular, boundaries of polytopes. Denote by B^d the closed unit ball at the origin in $\mathbb{R}^d$. A (disjoint) countable partition $\mathcal{P}$ of a topological space $\mathscr{M}$ is called a *semiregular cell-partition* if (1) each element $C \in \mathcal{P}$, called a cell of $(\mathscr{M}, \mathcal{P})$, is homeomorphic to $\operatorname{int} B^{\dim C}$, where $\dim C \in \mathbb{N}$ and $\dim C \leq \dim \mathscr{M}$; (2) the closure $\overline{C}$ (in $\mathscr{M}$) of each $C \in \mathcal{P}$ is the union of C and cells of smaller dimensions; (3) for each $C \in \mathcal{P}$ there is a mapping $r_C : \overline{C} \to B^{\dim C}$ which is a homeomorphism onto $r_C(\overline{C})$ and whose restriction to C is a homeomorphism onto $\operatorname{int} B^{\dim C}$. Authors that prefer to deal with closed cells refer to cell-partitions as *cell-complexes.*

If each cell is contained in the closure of finitely many cells, the partition is called *star-finite.* $\operatorname{Star} F$ denotes the subcomplex that consist of all (relatively open) cells whose closure contains F. If the closure of each cell is the union of finitely many cells, the partition is called *closure-finite.* When a partition is both closure- and star- finite, it is called *locally-finite.* Often, in the definition of cell-partition (cell-complex) one insists on that each closed cell is the image of a closed ball, which forces the compactness for each closed cell – we do not make such a requirement. Hence, in our definition the closures of the cells can be "semiclosed-semiopen". Our notion of semiregular cell-partition is a natural generalization of the standard notion of regular cell-partition, also known as regular CW-complex, introduced by J.H.C. Whitehead (see e.g. Ziegler, 2002). Namely, a regular cell-partition is a semiregular locally-finite cell-partition, where the closure of each cell is homeomorphic to a closed ball.

According to our definition, for example, the vertical projection on the plane of the graph (in $\mathbb{R}^3$) of a continuous piecewise-linear (more formally, piecewise-affine) function f on $\mathbb{R}^2$, which is defined by finitely many affine equations and inequalities, naturally induces a semiregular cell-partition $\mathcal{P}_f$ of $\mathbb{R}^2$: each 2-cell is a maximal linearity set of the function. However, $\mathcal{P}_f$ fails to be a regular cell-partition due to its unbounded cells. Semiregular partitions are especially well-suited for the study of topology of real semialgebraic and subanalytic sets – any such set has a canonical finite semiregular cell-partition.

It is easy to see that any regular cell-partition can be subdivided into a triangulation. Thus, a manifold admitting a regular cell-partition belongs to the category of PL-manifolds. It can be proven that a semi-regular locally-finite cell-partition can also be subdivided into a triangulation. However, since the algorithmic part of this paper deals only with finite partitions, we omit this theorem.

With any semiregular cell-partition there is a natural structure of poset. Namely, for cells F and C we have $F \preceq C$ if and only if $F \subset \overline{C}$; we write $F \prec C$ if $F \subset \overline{C}$ but $F \neq C$. If $F \preceq C$, we say that F is a face of C. We will use the same symbol $\mathcal{P}$ for a partition and its poset. It is convenient to augment the poset $\mathcal{P}$ with an infinum $\perp$, which is $\varnothing$, and a supremum $\top$, which can be thought of as all of $\mathscr{M}$. We use $\mathrm{Sk}_d(\mathscr{M}, \mathcal{P})$ to denote the d-skeleton of $(\mathscr{M}, \mathcal{P})$ - i.e. the subcomplex of $(\mathscr{M}, \mathcal{P})$ that consists of all cell of dimensions not exceeding d.

A subset of $\mathbb{R}^n$ is called polyhedral if it is defined by a propositional formula in the language of the reals ($\mathbb{R}$) that uses only affine equations and inequalities. A subset S of $\mathbb{S}^n \subset \mathbb{R}^{n+1}$ is called polyhedral if $S = \mathbb{S}^n \cap E$, where $E \subset \mathbb{R}^{n+1}$ is defined by a propositional formula in the language of the reals ($\mathbb{R}$) that uses only linear (homogeneous) equations and inequalities. A *PL-hypersurface* in $\mathbb{X}^n$ is a triple $(\mathscr{M}, \mathcal{P}; r)$, where $\mathscr{M}$ is a manifold with a semiregular cell-partition $\mathcal{P}$, and $r : \mathscr{M} \to \mathbb{X}^n$ is a continuous *realization* map, such that for each $C \in \mathcal{P}$ the set $r(\overline{C})$ is polyhedral and is homeomorphic to $\overline{C}$. Note that although r need not even be an immersion, the restriction of r to the closure of any cell C of $\mathscr{M}$ must be an embedding. The realization is called cellwise-flat if the dimension of the affine span of $r(\overline{C})$ is equal to $\dim C$. Although the term *face* is used both for abstract cells and their geometric realizations, we usually apply it for the realizations. If $\dim C = k$, then $r(C)$ is called a k-face of $(\mathscr{M}, \mathcal{P}; r)$. Throughout the paper all faces, just as all cells in topological partitions, are assumed to be *relatively open.* At times we refer to $(n-1)$-faces as *facets,* $(n-2)$-faces as *ridges,* and $(n-3)$-faces as *corners.* We may also use these geometric names for the underlying cells of $(\mathscr{M}, \mathcal{P})$ – the meaning will always be clear from the context.

Let us consider a connected *PL-hypersurface* $(\mathscr{M}, \mathcal{P}; r)$, where $r : \mathscr{M} \to \mathbb{R}^n$ maps each cell onto a set of the same affine dimension, i.e., $\dim C = \dim \mathrm{aff}\, r(C)$. Suppose $r : \mathscr{M} \to \mathbb{R}^n$ has at least one point of strict convexity; also, suppose that r is locally convex at all points of all corners of $(\mathscr{M}, \mathcal{P})$. Notice that if the last condition holds for some point of a corner, it holds for all points of the corner. Our main Theorem 8 states that under these conditions r is a homeomorphism on the boundary of a convex body. This theorem implies a test for global convexity of a PL-hypersurface that proceeds by checking the local convexity on each of the corners. The pseudo-code for the algorithm is given in Section 6.3. The complexity of this test depends not only on the model of computation, but also on the way the surface is given as input data. Let the input be the poset of faces of dimensions $n-1$ (facets) $n-2$ (ridges) and $n-3$ (corners). Suppose for each corner-ridge incidence (C, R) we are given a Euclidean inner normal to $r(R)$ at $r(C)$, and for each corner-facet incidence (C, F) we are given a Euclidean inner normal to $r(F)$ at $r(C)$. If we adopt the algebraic complexity model where each of scalar operations {comparison, addition, subtraction, multiplication, and division} has unit cost, the complexity of the algorithm is $O(n\mathsf{f}_{n-3\,n-2}) = O(n\mathsf{f}_{n-3\,n-1})$. Complexity under other models is studied in Section 7; we also study the complexity of extracting the required input information from more common input representations.

The algorithm consists of f_{n-3} independent subroutines corresponding to the $(n-3)$-faces, each with complexity not exceeding big-O in the number of $(n-2)$-cells incident to the $(n-3)$-face. In addition to the algorithmic implications, our generalization implies that any $(n-3)$-simple compact PL-hypersurface in $\mathbb{R}^n$ is the boundary of a convex polytope.

3 Geometry of Locally-Convex Immersions

Recall that a path joining points x and y in a topological space $\mathcal{T}$ is a continuous map $p : [0,1] \to \mathcal{T}$. Such a path is called an *arc* if p is injective. Denote by $\mathrm{Arcs}_{\mathscr{M}}(x, y)$ the set of all arcs joining $x, y \in \mathscr{M}$.

An immersion $i : \mathscr{M} \to \mathbb{X}^n$ induces a metric d_i on $\mathscr{M}$ by

$$d_i(x, y) = \inf_{a \in \mathrm{Arcs}_{\mathscr{M}}(x,y)} |i(a)|$$

where $|i(a)| \in \mathbb{R} \cup \infty$ stands for the length of the i-image of an arc a joining x and y on $\mathscr{M}$. This metric is called the i-metric. Of course, for a general continuous realization r it is not clear *a priori* that there is a path of finite length on $r(\mathscr{M})$ joining $r(x)$ and $r(y)$. That is why we need Lemma 1.

Lemma 1 *If i is an immersion, then any two points of $\mathscr{M}$ can be connected by an arc of a finite length. In particular, $\mathscr{M}$ is not only connected, but also arcwise connected.*

The following lemma is used implicitely throughout our proofs and is also important for understanding Van Heijenoort and Jonker & Norman' arguments that we are employing.

Lemma 2 *If i is an immersion, then the metric topology defined by the i-metric is equivalent to the original topology on $\mathscr{M}$.*

Van Heijenoort's (1952) proofs of these two lemmas, given for $\mathbb{X}^n = \mathbb{R}^n$, work for $\mathbb{X}^n = \mathbb{S}^n$ without changes. Note that since i is an immersion, then for a "sufficiently small" subset $\mathcal{S}$ of $\mathscr{M}$ the map $i|_{\mathcal{S}}$ is a homeomorphism and, therefore, the topology on $\mathcal{S}$ that is induced by the metric topology of $\mathbb{X}^n$, is equivalent to the intrinsic topology of $\mathcal{S}$, i.e., the subspace topology. Thus, for sufficiently small subsets of $\mathscr{M}$ the three topologies considered in this section are equivalent. This fact will be used on numerous occasions without an explicit reference to the above lemmas. We will also need the following theorem.

Theorem 3 (Van Heijenoort, 1952) *If a complete locally convex immersion f of a connected manifold $\mathscr{M}$ (*$\dim\ \mathscr{M} = n-1$*) into $\mathbb{R}^n$ $(n \geq 3)$ has a point of strict convexity, then f is a homeomorphism onto the boundary of a convex body.*

Recall that a subset C of the linear space $\mathbb{R}^{n+1}$ is called a *cone* if $\lambda C \subset C$ for any $\lambda \in \mathbb{R}_+$. A cone C is called pointed if $\mathbf{0} \subset C$. A cone is called salient if it does not contain any linear subspace except for $\mathbf{0}$. If $S \subset \mathbb{R}^{n+1}$, then we denote by $p \cdot S$ the cone with apex p over S.

Let x be a point on $\mathscr{M}$ and let S be a subset of $\mathbb{X}^n$ such that $x \cap r^{-1}(S) \neq \varnothing$ for a realization map $r : \mathscr{M} \to \mathbb{X}^n$. We denote by $r_x^{-1}S$ the connected component of $r^{-1}(S)$ that contains x. Consider now the case of $\mathbb{X}^n = \mathbb{S}^n$. We assume that $\mathbb{S}^n$ is embedded as the standard unit sphere into $\mathbb{R}^{n+1}$. For a point $a \in \mathbb{S}^n \subset \mathbb{R}^{n+1}$ we denote by c_a the central projection mapping from the half of $\mathbb{S}^n$ that is centered at a onto the tangent plane $\mathbf{T}_a \subset \mathbb{R}^{n+1}$ to $\mathbb{S}^n$ at a. To simplify the visual appearence of formulas we will use c_x instead of $c_{r(x)}$ in a context where the map $r : \mathscr{M} \to \mathbb{S}^n$ is fixed.

The following theorem, whose proof follows the approach taken by Jonker & Norman (1972), shows that for spherical immersions absence of a point of a strict convexity cannot result in the loss of global convexity, as it happens in the Euclidean case.

Theorem 4 *Let $i : \mathscr{M} \to \mathbb{S}^n$ $(n \geq 3)$ be a locally convex complete immersion of a connected $(n-1)$-manifold $\mathscr{M}$. Then $(\mathscr{M}, i)$ is strictly locally convex in at least one point, or $i(\mathscr{M}) = \partial(\mathbb{S}^n \cap C)$, where C is a pointed convex cone in $\mathbb{R}^{n+1}$.*

In the latter case we say that i is *conical.*

Proof. If $\mathscr{M}$ has a point p such that $i(p)$ is an extreme point for some convex witness K_p, then, by a classical theorem of Straszewicz (e.g. Rockafellar, 1990), either $i(p)$ is also an exposed point, or there are infinitely many exposed points of K_p arbitrarily close to $i(p)$. Since an exposed point is a point of strict convexity, the surface $(\mathscr{M}, i)$ is strictly convex in at least one point and the theorem follows.

Otherwise, suppose that for all $p \in \mathscr{M}$ the image $i(p)$ is an interior point of a segment on $i(\mathscr{M})$. Below we will show that this assumption implies that i is a homeomorphism onto $\partial(\mathbb{S}^n \cap C)$ for some pointed convex cone C. If a segment through $i(p)$ cannot be extended to a circle on $i(\mathscr{M})$, which is the i-image of a closed curve through p on $\mathscr{M}$, then pick an end point of a maximal segment I_p through $i(p)$ and call it $i(a)$. Since $i(a)$ is not extreme, it must lie in the interior of another segment. There is a support plane H_a for a such that $\operatorname{int} I_p$ and H_a do not intersect near $i(a)$. Consider now $i_a^{-1}H_a$, which is either a closed submanifold of dimension at least 1 or a closed submanifold with boundary. If $i_a^{-1}H_a$ is a submanifold with boundary, then the central projection $c_a i(i_a^{-1}H_a)$ of $i(i_a^{-1}H_a)$ on $\mathbf{T}_a \cong \mathbb{R}^n$ must have parallel lines on the boundary, which correspond to intersecting half-circles on $\mathbb{S}^n$. Furthermore, through any two points on the relative boundary of $c_a i(i_a^{-1}H_a)$ there are two parallel lines contained in the relative boundary of $c_a i(i_a^{-1}H_a)$. Let $R \subset \mathbf{T}_a$ be a maximal subspace contained in $c_a i(i_a^{-1}H_a)$. The c_a-preimages of points at infinity of R form a flat closed submanifold of $\mathscr{M}$. Thus, $\mathscr{M}$ must have a flat closed submanifold of dimension at least 1.

Let $\mathscr{F} \subset \mathscr{M}$ be a closed flat submanifold of maximal dimension. To simplify the visual appearance of formulas, denote by F the i-image of $\mathscr{F}$, and by $F_y^\perp$ the orthogonal complementary subspace to F in $\mathbb{S}^n$ at the point $r(y)$. If $\dim \mathscr{F} = n-1$, then $\mathscr{F} = \mathscr{M}$ and the proof is completed.

Let us turn now to the case of $\dim \mathscr{F} < n-1$. Since $(\mathscr{M}, i)$ is locally convex and does not have points of strict convexity, there is an open set $\mathscr{U}_{\mathscr{F}} \subset \mathscr{M}$ that contains $\mathscr{F}$ such that $i(\mathscr{U}_{\mathscr{F}})$ is of the form $\partial(\mathbf{0}\cdot F \times C_F) \cap \mathbb{S}^n$, where C_F is a salient $(n-1-\dim\mathscr{F})$-dimensional convex cone in $\mathbf{0}\cdot F_y^\perp$. Note that $i_y^{-1}(F_y^\perp \cap i(\mathscr{M}))$ is a locally convex hypersurface in $F^\perp$, which is strictly convex at y. If $\dim F^\perp = 2$, then i is conical – the pointed cone C is the product of a linear $(n-1)$-subspace in $\mathbb{R}^{n+1}$ and two rays (with origin at $\mathbf{0}$) in a complimentary linear 2-subspace.

Suppose now $\dim F_y^\perp > 2$. Upon applying van Heijenoort's theorem to the map $c_y i : i_y^{-1}(F_y^\perp \cap i(\mathscr{M})) \to c_y(F^\perp)$, we see that $c_y i\left[i_y^{-1}(F^\perp \cap i(\mathscr{M}))\right]$ is a complete convex hypersurface in $c_y(F_y^\perp) \subset \mathbf{T}_y$. But this is true for all $y \in \mathscr{F}$. Furthermore, if $y, y' \in \mathscr{F}$ are sufficiently close, the local convexity implies that $M_y = i\left[i_y^{-1}(F_y^\perp \cap i(\mathscr{M}))\right]$ and $M_{y'} = i\left[i_{y'}^{-1}(F_{y'}^\perp \cap i(\mathscr{M}))\right]$ are isometric: an isometry can be chosen as a minimal rotation $\rho_{yy'}$ of $\mathbb{S}^n$ in $\mathbb{R}^{n+1}$ around the orthogonal complement to the plane spanned by vectors $r(y)$ and $r(y')$, which maps $r(y)$ to $r(y')$. Since $\mathscr{F}$ is compact, the manifolds M_y are isometric for all values of $y \in \mathscr{F}$. Also, for any $y, y' \in \mathscr{F}$ the isometry mapping $\rho_{yy'}$ from M_y to $M_{y'}$ preserves the distance of each point of M_y to F. If the map $i : i_y^{-1}(F_y^\perp \cap i(\mathscr{M})) \to F_y^\perp$ is strictly convex at some point $z \neq y$, then any geodesic segment $I_z = i(\mathscr{I}_z)$, where $\mathscr{I}_z$ is a flat open 1-submanifold through $z \in \mathscr{M}$, is transversal to $F_y^\perp$. Therefore, any support plane H_{I_z} for $(\mathscr{M}, i)$ through I_z intersects with a support plane for $(\mathscr{M}, i)$ through F over a proper subset of F. But then for any neighborhood $\mathscr{U}_y$ of y there will be $y' \in \mathscr{U}_y \cap \mathscr{F}$ such that $\rho_{yy'}$ will move $r(z)$ to a point lying on the other side of H_{I_z} with respect to the convex witness of z, resulting in a contradiction. Thus, for all $y \in \mathscr{F}$ the hypersurface $c_y i\left[i_y^{-1}(F^\perp \cap i(\mathscr{M})) \subset c_y(F_y^\perp)\right]$ cannot be strictly convex at any $z \neq y$, which means $c_y i\left[i_y^{-1}(F^\perp \cap i(\mathscr{M}))\right]$ is a convex salient cone in $c_y(F_y^\perp)$. This implies that i is conical. ■

The notion of convex part, introduced by Van Heijenoort (1952), happens to be very useful in working with local convexity. A *convex part* of $(\mathscr{M}, i)$ at a point of strict convexity $i(o)$ (i.e. containing a point of strict convexity $i(o)$), is a connected subset C of $i(\mathscr{M})$, with $i_o^{-1}C$ open in $\mathscr{M}$, such that

(1) $i(\partial i_o^{-1}C) = H \cap i(\mathscr{M})$, where H is some hyperplane in $\mathbb{X}^n$, not passing through $i(o)$,

(2) C lies on the boundary of a convex body K_C such that $\partial K_C \subset C \cup H$.

The set $i_o^{-1}C$ is called an *abstract convex part at o* and is denoted by $\mathcal{C}$. We will denote $i(\partial i_o^{-1}C)$ by $\operatorname{rel}\partial C$ and we will call $H \cap K_C$ the *lid* of the convex part C. Let H_o be a supporting hyperplane at $i(o)$. Let us call the *open* half-space defined by H_o that contains $C \setminus i(o)$ the *positive half-space* and denote it by H_o^+.

Theorem 5 *(after van Heijenoort) If an immersion of an $(n-1)$-manifold ($n \geq 3$) into $\mathbb{X}^n$ has a point of strict convexity, it has a convex part containing this point.*

Proof. van Heijenoort's proof works for $\mathbb{X}^n$, $n \geq 3$, without changes. ■

Let us fix a supporting hyperplane H_o and consider a family of hyperplanes such that: (1) they are pairwise disjoint in H_0^+ and their intersections with H_0^+ form a partition of H_0^+, (2) they are all orthogonal to a line (circle) L, transversal to H_o, (3) they do not contain $i(o)$. Let us call such a family the fiber bundle $\{H\}_{(L,H_o)}$ of the positive half-space defined by L and H_o. Denote by $\lambda > 0$ the length of the segment on *the line (circle) L* from $l = L \cap H_o$ to $H \in \{H\}_{(L,H_o)}$, called the height of H. The elements of $\{H\}_{(L,H_o)}$ can then be indexed by their heights. We use H_λ for the hyperplane of height λ. For $\mathbb{X}^n = \mathbb{R}^n$ the range of λ is $(0, +\infty)$ and for $\mathbb{X}^n = \mathbb{S}^n$ it is $(0, \pi)$. If there exists a convex part at $i(o)$, then $\{H_\lambda\}_{(L,H_o)}$ defines a family $\{C_\lambda\}$ of convex parts at $i(o)$, ordered by inclusion. Each such convex part is "squeezed" between H_o and H_λ and inherits the height from its H_λ. Let us now consider the union $\boldsymbol{C}_o$ of all convex parts at $i(o)$: *we want to prove that $\boldsymbol{C}_o$ is a convex part itself.* This statement was proven by van Heijenoort for $\mathbb{X}^n = \mathbb{R}^n$ ($n \geq 2$) and we will prove it for $\mathbb{S}^n$ ($n \geq 3$).

Theorem 6 *Let $i : \mathscr{M} \to \mathbb{S}^n$ be as in Theorem 4, and let $C = i(\mathcal{C})$ be a convex part at $i(o)$, defined by a hyperplane $H_\lambda \subset \mathbb{S}^n$ ($\lambda \neq 0$) from a fiber bundle $\{H_\lambda\}_{(L,H_o)}$, where $L \perp H_o$. Suppose $\operatorname{rel}\partial C$ is the boundary of a convex set S in H_λ. Then either S is the i-image of a topological disk $\mathcal{S}$ ($\dim \mathcal{S} \leq n-1$) in $\mathscr{M}$ and $\mathscr{M} = \mathcal{C} \cup \mathcal{S}$, or C is a proper subset of a larger convex part at $i(o)$ and defined by the same bundle $\{H_\lambda\}_{(L,H_0)}$.*

Proof. We will now prove this theorem by a perturbation argument, which reduces the spherical case to that of $\mathbb{R}^n$. Since $\operatorname{rel}\partial S = \operatorname{rel}\partial C$, $S \subset H_\lambda$ ($\lambda > 0$), and $\dim S = n-1$, we conclude that $\overline{\operatorname{conv} C} \cap H_o = i(o)$. Since, by Lemma 1, $\mathscr{M}$ is arcwise connected, all of $\overline{\operatorname{conv} C}$, except for $i(o)$, lies in the positive halfspace H_o^+. Thus, there is a hyperplane H in $\mathbb{S}^n$, arbitrarily close to H_o and orthogonal to L, such that C lies in an open halfspace H^+ defined by H. Let p be the pole of $\mathbb{S}^n$ with respect to H that lies in H_o^+, and let $c_p : H^+ \to \mathbf{T}_p$ be the central projection map on the tangent plane $\mathbf{T}_p \subset \mathbb{R}^{n+1}$. The set $i_o^{-1}(H^+ \cap \mathscr{M})$ is obviously a manifold. The map $c_p i$ is a locally-convex immersion of $i_o^{-1}(H^+ \cap \mathscr{M})$ into $\mathbf{T}_p$ with a point of strict convexity, $i(o)$. Any Cauchy sequence on $\mathscr{M}$ under the $c_p i$-metric is also a Cauchy sequence under the i-metric. Thus the immersion map $c_p i : i_o^{-1}(H^+ \cap \mathscr{M}) \to \mathbf{T}_p$ is complete and, therefore, satisfies the conditions of Theorem 5. It maps any (spherical) convex part centered at $i(o)$ onto a Euclidean convex part; it also maps the fiber bundle $\{H_\lambda\}_{(L,H_o)}$ to a fiber bundle in $\mathbf{T}_p$. In the case of $\mathbb{X}^n = \mathbb{R}^n$ the statement of the theorem is known (van Heijenoort). We conclude that either $\mathscr{M} = \mathcal{C} \cup \mathcal{S}$, or C is a proper subset of a larger convex part centered at $i(o)$ and defined by the same bundle $\{H_\lambda\}_{(L,H_0)}$. ■

4 From Local to Global Convexity on the Sphere

Let us recall that Van Heijenoort proved that a complete locally convex immersion f of a connected manifold $\mathscr{M}$ (dim $\mathscr{M} = n-1$) into $\mathbb{R}^n$ ($n \geq 3$) is the boundary of a convex body, if f has a point of strict convexity. For $n = 3$ this result, according to van Heijenoort, follows from four theorems in Alexandrov's book (1948). Jonker & Norman (1973) proved that if f does not have a point of strict convexity, $f(\mathscr{M})$ is the direct affine product of a plane locally convex curve and a subspace $L \cong \mathbb{R}^{n-2}$ of $\mathbb{R}^n$.

Theorem 7 *Let $i : \mathscr{M} \to \mathbb{S}^n$ ($n \geq 2$) be an immersion of a connected $(n-1)$-manifold $\mathscr{M}$, satisfying the following conditions: 1) $\mathscr{M}$ is complete with respect to the i-metric, 2) $(\mathscr{M}, i)$ is locally convex at all points of $\mathscr{M}$. Then $i : \mathscr{M} \to \mathbb{S}^n$ is an embedding onto the boundary of a convex body in $\mathbb{S}^n$.*

Proof. If points of strict convexity are absent, then by Theorem 4 the map $i : \mathscr{M} \to \mathbb{S}^n$ is an embedding onto the boundary of a convex body. Thus, we can assume that a point of strict convexity exists. If $o \in \mathscr{M}$ is such a point, then by Theorem 5 there is a convex part containing $i(o)$. Consider the union $\boldsymbol{C}_o$ of all convex parts at $i(o)$. Denote by ζ the least upper bound of the heights of convex parts defined by $\{H_\lambda\}_{(L,H_o)}$ at $i(o)$. By Theorem 6 $\partial \boldsymbol{C}_o \subset H_\zeta \cap i(\mathscr{M})$ and rel $\partial \boldsymbol{C}_o$ is the boundary of a closed convex set D in H_ζ. Two mutually excluding cases are possible.

`Case 1:` $\dim D < n-1$. Then, following the argument of van Heljenoort (Part 2: pp. 239-230, Part 5: p. 241, and Part 3: II on p. 231), we conclude that $\boldsymbol{C}_o \cup D$ is the homeomorphic i-image of the $(n-1)$-sphere $i^{-1}(\boldsymbol{C}_o) \cup i^{-1}(D) \subset \mathscr{M}$, where $i^{-1}(\boldsymbol{C}_o)$ is a maximal abstract convex part at o. Since $\mathscr{M}$ is connected, $i^{-1}(\boldsymbol{C}_o) \cup i^{-1}(D) = \mathscr{M}$, and $i : \mathscr{M} \to \mathbb{S}^n$ is a convex embedding of $\mathscr{M}$.

`Case 2:` $\dim D = n-1$. By Theorem 6 $\boldsymbol{C}_o$ is either a proper subset of a larger convex part, or $\boldsymbol{C}_o$, together with the lid D, is the homeomorphic i-image of $\mathscr{M}$. Since the former alternative is excluded by definition of $\boldsymbol{C}_o$, $i : \mathscr{M} \to \mathbb{S}^n$ is a convex embedding onto $\partial(\boldsymbol{C} \cup D)$. ■

Note that the statement of the above theworem is invalid for $n = 2$. For example, although the 1-surface in $\mathbb{S}^2$ depicted in Figure 1 is locally convex at all points, it does not bound any convex set on $\mathbb{S}^2$.

5 Locally convex PL-surfaces

Let $\mathcal{P}$ be a fixed star-finite semi-regular cell-partition of $\mathscr{M}$. Recall that in our terminology a cell is always homeomorphic to an open ball. We say that r is locally convex at a cell $C \in \mathcal{P}$ if it is locally convex at each point of C.

Theorem 8 *Let $r : \mathscr{M} \to \mathbb{R}^n$ ($n > 2$) be a complete cellwise-flat PL-realization of a connected manifold $\mathscr{M}$* ($\dim \mathscr{M} = n-1$) *such that*

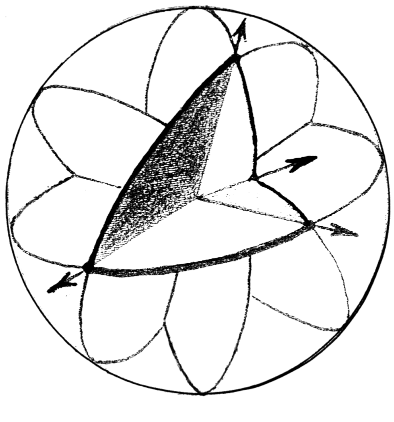

Fig. 1. Locally-convex PL-hypersurface in $\mathbb{S}^2$, which is not convex

1) r is locally convex in at least one point of each $(n-3)$-cell.

2) r is bounded or strictly locally convex in at least one point of $\mathscr{M}$.

Then $r : \mathscr{M} \to \mathbb{R}^n$ is an embedding onto the boundary of a convex body defined by (possibly infinitely many) affine inequalities.

Proof. Upon invoking the definition of cellwise-flat PL-realization and that of local convexity, we conclude that r is an immersion. We know that $r : \mathscr{M} \to \mathbb{R}^n$ is locally convex at all $(n-3)$-cells. Since r is a cellwise-flat PL-realization, r is also locally-convex at all d-cells for $d > n-3$. If r is bounded, then $\operatorname{conv} r(\mathscr{M})$ is bounded and compact. By Straszewicz's theorem (Rockafellar, 1990), the set $\operatorname{conv} r(\mathscr{M})$ has an exposed point. As r is complete, this exposed point must the r image of some $x \in \mathscr{M}$. Since r is locally convex at x, it is also strictly locally convex at x. If we prove that r is locally convex at all cells, by Theorem 3 the map $r : \mathscr{M} \to \mathbb{R}^n$ is a convex embedding. We proceed by reverse induction in cell's dimension. Suppose we have shown that $r : \mathscr{M} \to \mathbb{R}^n$ is locally convex at each k-cell, where $0 < k \leq n-3$ (and therefore at all cells of higher dimensions). If $n-3=0$, the proof is finished. So, let $n \geq 4$ and let us consider a $(k-1)$-cell $F \in \mathcal{P}$. Consider $r(\operatorname{Star} F) \cap \mathbb{S}_F$, where $\mathbb{S}_F$ is a sufficiently small $(n-k)$-sphere lying in an affine subspace complementary to $\operatorname{aff} r(F)$ and centered at some point of $r(F)$. Also note that $\dim \mathbb{S}_F = n-k \geq 2$. The map $r : \mathscr{M} \to \mathbb{R}^n$ is locally convex at F if and only if the hypersurface $\mathbb{S}_F \cap r(\operatorname{Star} F)$ in $\mathbb{S}_F$ is convex. Since $r : \mathscr{M} \to \mathbb{R}^n$ is locally convex at each k-cell, the surface $\mathbb{S}_F \cap \operatorname{Star} F$ is locally convex at each vertex and therefore locally convex everywhere. The set $r^{-1}(\mathbb{S}_F \cap r(\operatorname{Star} F))$ is complete in the r-metric and thus, by Theorem 7, the surface $\mathbb{S}_F \cap r(\operatorname{Star} F)$ is an embedded convex hypersurface in $\mathbb{S}_F$. Notice that the condition $\dim \mathbb{S}_F = n-k > 2$ is essential to the applicability of Theorem 7 (see Figure 1 for a locally convex surface in $\mathbb{S}^2$ which is not a convex surface in $\mathbb{S}^2$). So, $r : \mathscr{M} \to \mathbb{R}^n$ is locally convex at F.

To recap, the above induction argument shows that r is locally convex at all vertices, and, therefore, at all points. The metric induced by r is indeed complete. Upon applying Van Heijenoort's Theorem 3 and Theorem 7, we conclude that r is an embedding and that $r(\mathscr{M})$ is the boundary of a convex body. ■

Corollary 9 *Let $r : \mathscr{M} \to \mathbb{R}^n$ $(n > 2)$ be a complete cellwise-flat PL-realization of a connected manifold $\mathscr{M}$ $(\dim \mathscr{M} = n-1)$. Suppose r is bounded or is strictly locally convex in at least one point. If $(\mathscr{M}, \mathcal{P})$ is $(n-3)$-simple, i.e. exactly three $(n-1)$-cells make contact at each $(n-3)$-cell, then $r(\mathscr{M})$ is the boundary of a convex polyhedron.*

6 Convexity Checker for PL-hypersurfaces

In this section we present a polynomial-time algorithm for checking the convexity of any PL-realization $r : \mathscr{M} \to \mathbb{R}^n$ $(n \geq 3)$ of a semi-regular cell-partition $\mathcal{P}$ of a connected compact $(n-1)$-manifold $\mathscr{M}$. The map r under testing is assumed to be cellwise-flat (see Section 2), which implies that each cell $C \in \mathcal{P}$ is homeomorphicly mapped by r to an open subset of an affine subspace of dimension $\dim C$. We do not assume that the realization is an immersion: if it is not an immersion, the algorithm will detect this. We do not make any generic position assumptions.

In describing the algorithm we assume that certain combinatorial and geometric information is readily available. This input information is exactly what should be kept by a convex hull computer if it is to use our verification procedure. Later we discuss the complexity of extracting the necessary input information from PL-surface descriptions given in some typical formats.

If any of the subprocedures return 1 or "false", the main procedure returns "false" as the final answer. The idea of the algorithm is to check that the immersion and the local convexity properties hold at each corner. For each corner C this check is reduced, roughly speaking, to the verification of convexity of a certain cone $K(r, \operatorname{Star} C)$ in $r(C)^{\perp} \equiv \mathbb{R}^3$, which is constructed from the poset of $\operatorname{Star} C$ and the restriction of $r : \mathscr{M} \to \mathbb{R}^n$ to $\operatorname{Star} C$. Such a cone is not unique – for example, any non-singular affine transformation of $K(r, \operatorname{Star} C)$ is just as good as $K(r, \operatorname{Star} C)$. This reduction from the star of a corner to a cone in $\mathbb{R}^3$ is done by the procedure Reduce-to-3D.

6.1 Input Conventions

Let X be a subset of $\mathbb{R}^n$. We use $\operatorname{aff} X$ to denote the affine subspace spanned by X and $\overrightarrow{\operatorname{aff} X}$ to denote the *linear* subspace $\{x - x' \mid x, x' \in \operatorname{aff} X\}$. For a (not necessarily convex) polytope $\Pi \subset \mathbb{R}^n$ with a face F an *inner normal* at F is any vector $\boldsymbol{n}$ in $\overrightarrow{\operatorname{aff} \Pi}$ such that for any point p on F there is $\varepsilon > 0$ such that $p + \varepsilon \boldsymbol{n} \in \operatorname{int} \Pi$. The normal $\boldsymbol{n}$ is called a Euclidean normal if $\boldsymbol{n} \perp \operatorname{aff} F$.

As a reminder, we refer to the $(n-3)$- and $(n-2)$-cells of the partition as *corners* and *ridges* respectively, and we refer to $(n-1)$-faces as *facets*. We also use these terms to refer to the realizations of these cells in $\mathbb{R}^n$. Mathematically, the input is given as follows:

(1) the subposet $\mathcal{P}[n-3, n-2, n-1] \subset \mathcal{P}$ of corners, ridges, and facets where it is known in advance which are which;
(2) a Euclidean inner normal to R at C for each ridge-corner incidence (R, C);
(3) a Euclidean inner normal to F at C for each facet-corner incidence (F, C).

The data in (1) will be referred to as *combinatorial*, and that, described in (2) and (3), as *linear-algebraic*. We assume that each vector in the linear-algebraic data ”knows” the corresponding abstract cells in $\mathcal{P}$, and that each abstract cell in $\mathcal{P}[n-3, n-2, n-1]$ “knows” all normal vectors related to it. The input data-structure can be implemented as a double-linked adjacency list, with appropriate attribute fields for dimensional and linear-algebraic data. Namely, we can create an adjacency list for the directed (multi-) graph whose vertex set consists of elements of $\mathcal{P}[n-3, n-2, n-1]$ and whose edge set consists of all ordered pairs (C,C$'$) and (C’,C), where C $\prec$ C$'$ or C$'$ $\prec$ C in $\mathcal{P}[n-3, n-2, n-1]$. When the input is available in this form we say that the *input is given in the standard form.*

In applications a PL-hypersurface is usually specified by a subposet of the face poset, which includes the vertices or the facets or both; it is normally equipped either with the coordinates of vertices or with the equations (or unequalities) for the facets. Suppose now the input is given as the poset $\mathcal{P}[0, n-3, n-2, n-1]$, equipped with the coordinates of the vertices; in this case we will say that the input is given in *traditional* form. If the partition $\mathcal{P}$ is a triangulation, then the linear-algebraic data required for our algorithm (standard form) can be produced in linear time in $\mathsf{f}_{n-3\, n-2}$, which is also polynomial in the total bit size of the input. More generally, if the face numbers of facets of $(\mathscr{M}, \mathcal{P})$ are bounded by a universal constant (in $\mathsf{f}_{n-3\, n-2}$), the linear-algebraic data for the standard form of the input can be computed by using $O(\mathsf{f}_{n-2\, n-3})$ field arithmetic operations.

6.2 Preprocessing

By *preprocessing* in the context of problems of verification of geometric properties we mean any computation that does not depend on the geometric realization (in our case r), but only on the topology or combinatorics of the object (in our case – the pair $(\mathscr{M}, \mathcal{P})$).

Since $\mathscr{M}$ is a manifold, the facets of $(\mathscr{M}, \mathcal{P})$ making contact at a corner are “glued” to each other in a circular fashion. Same can be said about the ridges. The circular structure of the stars of $(n-3)$-cells implies that for each $(n-3)$-cell C we have $\mathsf{f}_{n-2}(\mathrm{Star}\, C) = \mathsf{f}_{n-3}(\mathrm{Star}\, C)$. The last formula implies that for the whole $\mathcal{P}$ we have $\mathsf{f}_{n-3\, n-2} = \mathsf{f}_{n-3\, n-1}$. More properly, a topologist would say that the “links” (defined via the 1-skeleton of the *dual partition*, a well-known construction going back to H. Poincare: see Seifert & Threlfall, 1980) of the corners are *circles.* These circles can be thought of as polygons whose vertices correspond to the facets of $\mathcal{P}$ and edges to the ridges of $\mathcal{P}$. To apply our algorithm we need to determine a cyclic order of ridges around each corner. Such an order is unique up to the choice of direction.. To apply our algorithm for different realizations of the same cell-partition of $\mathscr{M}$ it is reasonable to maintain a cyclic order of ridges around each corner; this can be considered as preprocessing.

6.3 Algorithm

The main procedure Convexity-Checker is given in Algorithm 1. Convexity-Checker works on a stack *Corners*, in which we put all $(n-3)$-faces of $(\mathscr{M}, \mathcal{P})$ prior to starting. Subroutines Reduce-to-3D and Check-if-Cone-Convex are used by Convexity-Checker, and subroutine Is-Folded is used by Reduce-to-3D. Before describing the working of the algorithm, let us define a couple of auxiliary notions.

For any $m \geq 2$ a graph with vertex set $V = \{0, 1, \ldots, m\}$ $(m \geq 2)$ and edge set $E = \{(01), \ldots, (0m)\} \cup \{(12), \ldots, (m-1\,m), (m, 1)\}$ is called the m-wheel graph and is denoted by W_m. Vertices $RimV(W_m) = \{1, \ldots, n\}$ are called rim vertices; edges $RimE(W_m) = \{(12), \ldots, (m-1\,m), (m, 1)\}$ are called rim ridges. Edges $\{(01), \ldots, (0m)\}$ are called spokes. Vertex 0 is called the center of W_m. Let $p : V \to \mathbb{R}^3$ be a realization of the vertex set of W_m in $\mathbb{R}^3$. If $m \geq 3$, then we can assign to each 3-cycle $(0\,i\,i+1)$ (where $i+1$ is taken $\mod m$) in W_m a geometric simplex in $\mathbb{R}^3$ with the vertices $p(0)$, $p(i)$, and $p(i+1)$. Therefore, the map $p : V \to \mathbb{R}^3$ produces a simplicial surface with boundary. With a slight abuse of terminology we will say that a realization $p : V \to \mathbb{R}^3$ of the wheel graph $W_m(V, E)$ (where $m \geq 3$) is convex if p is injective and the resulting simplicial 2-surface, which we denote by $p[W_m]$, is convex at $p(0)$. While we may encounter Euclidean realizations of the vertex set of W_2 , we will not have a need to associate a surface in $\mathbb{R}^3$ with such realizations.

Once a corner C is popped from the stack, a pair $(W_m, \mathbf{n}) = (W_m[C], \mathbf{n}[C])$ is created. This pair consists of the wheel graph $W_m = W_m[C]$, which encodes the combinatorics of $\operatorname{Star} C$, and an array of vectors $\mathbf{n}[C]$, whose elements are the inner normals to the r-realizations of ridges and facets of $\operatorname{Star} C$ at the face $r(C)$. Namely, m is the number of ridges meeting at C, the center of W_m corresponds to C, the rim vertices correspond to the ridges of $\operatorname{Star} C$, the rim edges correspond to the facets of $\operatorname{Star} C$, and the "spokes" correspond to the corner-ridge incidences.

Algorithm 1 Convexity-Checker

```
while $Corners \neq \varnothing$ do
    Pop an $(n-3)$-face $C$ from $Corners$
    Create $(W_m, \mathbf{n})$ for Star $C$
    if Reduce-to-3D$(W_m, \mathbf{n}) = 1$ then return $false$                ▷ not convex
    else
        if Reduce-to-3D$(W_m, \mathbf{n}) \neq 0$ then
            $(W_m, p) \leftarrow$ Reduce-to-3D$(W_m, \mathbf{n})$
            if Check-if-Cone-Convex-in-3D$(W_m, p)$=$false$ then return $false$   ▷ not convex
            end if
        end if
    end if
end while
    return $true$                                                              ▷ yes, convex
```

6.4 Reduce-to-3D *and its Subroutines*

For a corner C of $(\mathscr{M}, \mathcal{P})$ let $\{0, 1, \ldots, m\}$ be the vertices of the wheel graph $W_m = W_m[C]$, with 0 as the center of W_m (which corresponds to the conrer C), and let $[1, \ldots, m]$ be a cyclically ordered list of the rim vertices of W_m (which correspond to the ridges of $\operatorname{Star} C$). The input to Reduce-to-3D consists of: the wheel graph W_m; the Euclidean inner normals $\boldsymbol{n}_1, ..., \boldsymbol{n}_m$ for the r-images of ridges of $\operatorname{Star} C$ at $r(C)$; the Euclidean inner normals $\boldsymbol{n}_{12}, ..., \boldsymbol{n}_{m1}$ for the r-images of the facets of $\operatorname{Star} C$ at $r(C)$. The arrays of normal vectors can also be thought of as a map $\mathbf{n} : RimE(W_m) \cup RimV(W_m) \to \mathbb{R}^n$.

Reduce-to-3D uses a subroutine Is-Folded, which checks for violations of the immersion assumption in cases where all normals are in the same plane. The output of Reduce-to-3D is either one of $\{0, 1\}$ or a realization of W_m in $\mathbb{R}^3$, where the center of W_m is mapped to the origin. Output 1 means that the input $(W_m, \mathbf{n})$ is inconsistent with our assumptions about the integrity of the input data or the immersion assumptions. Output 0 means that $\operatorname{Star} C$ has passed the local convexity check.

6.4.1 Is-Folded

Let $\boldsymbol{v}, \boldsymbol{u}, \boldsymbol{w}$ be three coplanar non-zero vectors. The ordered triple $(\boldsymbol{v},\ \boldsymbol{u},\ \boldsymbol{w})$ defines a plane angle at the origin in the following way: $\boldsymbol{v}$ and $\boldsymbol{w}$ span the two extreme rays of the angle, while $\boldsymbol{u}$ is an interior vector of the angle – i.e., the function of $\boldsymbol{u}$ is to specify which of the two open subsets defined by $\boldsymbol{v}$ and w is interior to the angle. We denote such angle by $\langle \boldsymbol{v}|\boldsymbol{u}|\boldsymbol{w}\rangle$. Note that $\langle \boldsymbol{v}|\boldsymbol{u}|\boldsymbol{w}\rangle = \langle \boldsymbol{w}|\boldsymbol{u}|\boldsymbol{v}\rangle$.

Is-Folded takes as input a 5-tuple of coplanar non-zero vectors $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e})$, where $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$ are pairwise distinct and $\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e}$ are pairwise distinct. Is-Folded returns *true* if the interiors of angles $\langle \boldsymbol{a}|\boldsymbol{b}|\boldsymbol{c}\rangle$ and $\langle \boldsymbol{c}|\boldsymbol{d}|\boldsymbol{e}\rangle$ overlap and *false* otherwise. For example, Figure 2 shows the case where "folding" takes place: angle $\langle \boldsymbol{c}|\boldsymbol{d}|\boldsymbol{e}\rangle$ "folds over" the angle $\langle \boldsymbol{a}|\boldsymbol{b}|\boldsymbol{c}\rangle$.

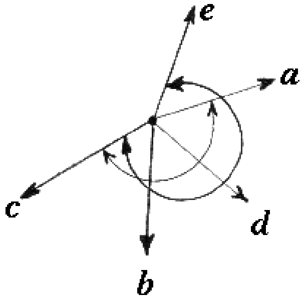

Fig. 2. Angles $\langle \boldsymbol{a}|\boldsymbol{b}|\boldsymbol{c}\rangle$ and $\langle \boldsymbol{c}|\boldsymbol{d}|\boldsymbol{e}\rangle$ overlap.

In the pseudocode of this procedure we will use a boolean predicate $P(\boldsymbol{v}|\boldsymbol{u}_1, \boldsymbol{u}_2|\boldsymbol{w})$, which is defined for any 4-tuple of coplanar vectors $\boldsymbol{v}, \boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{w}$, where $\boldsymbol{v}$ and $\boldsymbol{w}$ are distinct and

$\boldsymbol{u}_i \neq \boldsymbol{v}$, $\boldsymbol{u}_i \neq \boldsymbol{w}$ for $i = 1, 2$. $P(\boldsymbol{v}|\boldsymbol{u}_1, \boldsymbol{u}_2|\boldsymbol{w})$ is *false* if $\langle \boldsymbol{v}|\boldsymbol{u}_1|\boldsymbol{w}\rangle \neq \langle \boldsymbol{v}|\boldsymbol{u}_2|\boldsymbol{w}\rangle$ (Figure 3, right) and *true* otherwise (Figure 3, left).

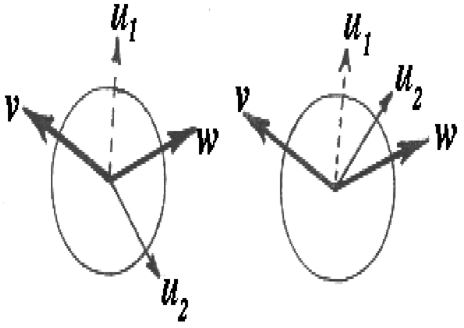

Fig. 3. Left: $P(\boldsymbol{v}|\boldsymbol{u}_1, \boldsymbol{u}_2|\boldsymbol{w}) = false$. Right: $P(\boldsymbol{v}|\boldsymbol{u}_1, \boldsymbol{u}_2|\boldsymbol{w}) = true$.

Input: $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e} \in \mathbb{R}^n$, where $\dim \operatorname{span}\{\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e}\} = 2$, $|\{\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}\}| = 3$, $|\{\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e}\}| = 3$

Output: boolean

```
Algorithm Is-Folded
  if $\boldsymbol{a}$ and $\boldsymbol{e}$ define the same ray then
      if $P(\boldsymbol{c}|\boldsymbol{b}, \boldsymbol{d}|\boldsymbol{a}) = true$ then return true
      else   return false
      end if
  end if
  if $P(\boldsymbol{a}|\boldsymbol{b}, \boldsymbol{e}|\boldsymbol{c}) = false$ and $P(\boldsymbol{c}|\boldsymbol{d}, \boldsymbol{e}|\boldsymbol{a}) = true$ and $P(\boldsymbol{b}|\boldsymbol{c}, \boldsymbol{d}|\boldsymbol{e}) = true$ then return false
  else   return true
  end if
```

The following algorithm shows how to compute $P(\boldsymbol{v}|\boldsymbol{u}_1, \boldsymbol{u}_2|\boldsymbol{w})$ via standard linear algebra. For any ordered pair of vectors $[\boldsymbol{e}_1, \boldsymbol{e}_2]$, such that $\{\boldsymbol{e}_1, \boldsymbol{e}_2\} \subset \operatorname{span}\{\boldsymbol{v}, \boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{w}\}$, we use $\operatorname{sgn}[\boldsymbol{e}_1, \boldsymbol{e}_2]$ to denote the orientation of $[\boldsymbol{e}_1, \boldsymbol{e}_2]$ with respect to some fixed orientation of $\operatorname{span}\{\boldsymbol{v}, \boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{w}\}$).

```
Algorithm $P(\boldsymbol{v}|\boldsymbol{u}_1, \boldsymbol{u}_2|\boldsymbol{w})$
    $s \leftarrow \operatorname{sgn}[\boldsymbol{v}, \boldsymbol{w}]$
    if $\operatorname{sgn}[\boldsymbol{u}_1, \boldsymbol{w}] = \operatorname{sgn}[\boldsymbol{v}, \boldsymbol{u}_1] = s$ then
        if $\operatorname{sgn}[\boldsymbol{u}_2, \boldsymbol{w}] = \operatorname{sgn}[\boldsymbol{v}, \boldsymbol{u}_2] = s$ then return true
        else   return false
        end if
    else
        if $\operatorname{sgn}[\boldsymbol{u}_2, \boldsymbol{w}] = \operatorname{sgn}[\boldsymbol{v}, \boldsymbol{u}_2] = s$ then return false
        else   return true
        end if
    end if
```

6.4.2 Procedure Reduce-to-3D

Input: W_m: m-wheel graph, $\mathbf{n} : RimE(W_m) \cup RimV(W_m) \rightarrow \mathbb{R}^n$. In the pseudocode we use $\boldsymbol{n}_i$ for $\mathbf{n}(i)$ and $\boldsymbol{n}_{i\,i+1}$ for $\mathbf{n}(i\,i+1)$.

Output: one of $\{0;\ 1;\ (W_m, p)$ where $p : V(W_m) \rightarrow \mathbb{R}^3\}$)

Algorithm 2 Reduce-to-3D

```
$\boldsymbol{e}_1 \leftarrow \boldsymbol{n}_1$, $\mathbf{S} \leftarrow \{\boldsymbol{e}_1\}$ // $\mathbf{S}$ is a maximal independent set of vectors in $(\overrightarrow{\mathrm{aff}\, r(C)})^{\perp}$
if rank$\{\boldsymbol{n}_1, \boldsymbol{n}_2\} = 1$ then
    if $\boldsymbol{n}_2 = \lambda \boldsymbol{n}_1$ for some $\lambda \geq 0$ then return 1
    end if
else $\boldsymbol{e}_2 \leftarrow \boldsymbol{n}_2$, $\mathbf{S} \leftarrow \mathbf{S} \cup \{\boldsymbol{e}_2\}$
    if $m = 2$ then
        if $\{\mathrm{rank}\{\boldsymbol{n}_1, \boldsymbol{n}_{12}, \boldsymbol{n}_{21}\} = 2\}$ and $\{\mathrm{sgn}[\boldsymbol{n}_1, \boldsymbol{n}_{12}] = \mathrm{sgn}[\boldsymbol{n}_1, \boldsymbol{n}_{21}]\}$ then return 1
        else   return 0
        end if
    end if
end if
$i \leftarrow 3$    //rim-vertex counter
while $i \neq 1 \mod m$ and $|\mathbf{S}| < 3$ do
    if rank$\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{n}_i\} \leq 2$ then    //$\boldsymbol{n}_i$, $\boldsymbol{e}_1, \boldsymbol{e}_2$ are all in one plane
        if $\boldsymbol{n}_i = \lambda \boldsymbol{e}_1$ or $\boldsymbol{n}_i = \lambda \boldsymbol{e}_2$ for some $\lambda > 0$ then return 1
        end if
        if $|\mathbf{S}| = 1$ then    //$\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ are collinear and contraoriented
            if $\mathrm{sgn}[\boldsymbol{e}_2, \boldsymbol{n}_{12}] = \mathrm{sgn}[\boldsymbol{e}_2, \boldsymbol{n}_{2m}]$ then return 1
            else
                if $m = 3$ then return 0
                else $\boldsymbol{e}_2 \leftarrow \boldsymbol{n}_3$, $\mathbf{S} \leftarrow \mathbf{S} \cup \boldsymbol{e}_2$
                end if
            end if
        else   //in this case we know $|\mathbf{S}| = 2$
            if Is-Folded$(\boldsymbol{e}_1, \boldsymbol{n}_{12}, \boldsymbol{e}_2, \boldsymbol{n}_{i-1\,i}, \boldsymbol{n}_i) = \mathit{true}$ then return 1
            end if
        end if
    else $\boldsymbol{e}_3 \leftarrow \boldsymbol{n}_i$, $\mathbf{S} \leftarrow \{\boldsymbol{e}_3\}$
    end if
    $i \leftarrow i + 1$
end while
if $i = 1 \mod m$ or $|\mathbf{S}| < 3$ then return 0
end if
$p(0) \leftarrow (0, 0, 0)$
for $j = 1$ to $m$ do
    $p(j) \leftarrow (\boldsymbol{e}_1 \cdot \boldsymbol{n}_j, \boldsymbol{e}_2 \cdot \boldsymbol{n}_j, \boldsymbol{e}_3 \cdot \boldsymbol{n}_j)$
end for
return $(W_m, p)$
```

6.5 *Procedure* Check-if-Cone-Convex

Denote by $p[W_m]$ the simplicial surface that results from the map $p : V(W_m) \to \mathbb{R}^3$ (see Subsection 6.3). Informally speaking, we test the surface $p[W_m]$ for convexity by going around the wheel and checking whether the following two conditions are satisfied or not. The first condition will be the conical PL-surface $p[W_m]$ "turning in the same direction" each time we increment the index: here, the formal meaning of "the same direction" is captured by the linear-algebraic notion of orientation of a frame. Recall that the sign of a list $[\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3]$ of three vectors in $\mathbb{R}^3$ is the sign of the determinant of the 3×3 matrix whose i-th row is $\boldsymbol{v}_i$, which we denote by $\operatorname{sgn}[\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3]$. If $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3 \in \mathbb{R}^n$ with $n > 3$, then to define the sign of the triple $[\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3]$ we need to fix an orientation in a 3-subspace of $\mathbb{R}^n$ containing $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3\}$. Let $\operatorname{sgn}[p(1), p(2), p(3)] = s \neq 0$. Formally, the first condition is that for every $i \in V(W_m)$ the sign $\operatorname{sgn}[p(i), p(i+1), p(i+2)]$ must be s or 0; the zero sign corresponds to the case where the surface "continues straight", i.e., vectors $p(i), p(i+1), p(i+2)$ lie in one plane. The second condition will be $p[W_m]$ not intersecting itself. This can be captured by checking the sign of every triple of the form $[p(1), p(i), p(i+1)]$ is 0 or $s = \operatorname{sgn}[p(1), p(2), p(3)]$. Thus, roughly speaking, the first condition ensures that the convexity is not lost due to a turn in the wrong direction, while the second condition guarantees the cone will not intersect itself. Finally, the correctness of Check-if-Cone-Convex hinges on the following lemma, whose proof we omit.

Lemma 10 *Let $p : V(W_m) \to \mathbb{R}^3$ be a realization of the m-wheel graph ($m \geq 3$) which does not map any two consecutive spokes into the same ray. Suppose $\operatorname{sgn}[p(1), p(2), p(3)] \neq 0$. Then $p[W_m]$ lies on the boundary of a convex cone if and only if for all $0 \leq i \leq m - 1$:*

1) $\operatorname{sgn}[p(i), p(i+1), p(i+2)] = \operatorname{sgn}[p(1), p(2), p(3)]$ *and*

2) $\operatorname{sgn}[p(1), p(i+1), p(i+2)] = \operatorname{sgn}[p(1), p(2), p(3)]$.

The input to Check-if-Cone-Convex-in-3D is wheel graph W_m ($m > 1$), each of whose vertices v is equipped with a corresponding point $p(v)$ in $\mathbb{R}^3$. For notational simplicity we assume that the vertex set $V(W_m)$ of W_m is $\{0, 1 \ldots, m\}$, where 0 is the center of W_m, and $[1, \ldots m]$ is a cyclic order on the rim vertices. In addition, we assume that *vertex* 0 *is realized at the origin* $\mathbf{0}$.

Input: W_m: wheel graph, $p : V(W_m) \to \mathbb{R}^3$

Output: boolean

Algorithm 3 Check-if-Cone-Convex

```
  $sign \leftarrow \mathrm{sgn}[\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3]$
  for $j$ from 2 to $m$ do
      if $\{\mathrm{sgn}[p(j), p(j+1), p(j+2)] = -sign\}$ or $\{\mathrm{sgn}[\boldsymbol{e}_1, \boldsymbol{e}_2, p(j+2)] = -sign\}$ then
          return $false$
      end if
  end for
  return $true$
```

7 Complexity Analysis

In this section f_k denotes the number of k-cells of $(\mathscr{M}, \mathcal{P})$, and f_{kl} the number of incidences between k-cells and l-cells of $(\mathscr{M}, \mathcal{P})$; if $\mathcal{P}'$ is a subposet of $\mathcal{P}$, then $\mathsf{f}_k(\mathcal{P}')$ denotes the number of k-cells in $\mathcal{P}'$. Note that all linear algebra in the algorithm is essentially reduced to comparisons of signs of lists of at most three n-vectors; we will refer to any such calculation as a *sign computation.* Unless mentioned otherwise, as e.g. in the next Subsection, we assume that the input is in the standard form.

(1) Building the wheel graph for $\mathrm{Star}\, C$ takes time linear in the number of ridges of $\mathrm{Star}\, C$.
(2) Since *Corners* is accessed at most f_{n-3} times, Reduce-to-3D is called at most f_{n-3} times.
(3) Reduce-to-3D$(W_m, \boldsymbol{n})$ requires at most $O(m)$ sign computations.
(4) Check-if-Cone-Convex requires at most $O(m)$ sign computations.
(5) Is-Folded requires a constant number of sign computations.

1 *Suppose the algorithm uses the field arithmetic $(+, -, \times, \div)$ and each arithmetic operation has unit cost.* This model is realistic when real computations are conducted with floating point arithmetic. If n is fixed, the complexity of the algorithm is $O(\mathsf{f}_{n-3\,n-2}) = O(\mathsf{f}_{n-3\,n-1})$. To estimate the complexity in the case where n is one of the parameters describing the input size, we need to estimate the contributions of *sign computations* in (3)–(5). Notice that any sign computation in (3)-(5) deals with, at most, six n-vectors. Since standard linear-algebraic procedures over a field can be used, the complexity of the algorithm is $O(n\mathsf{f}_{n-3\,n-2})$.

2 *What follows is a discussion of the complexity in the cases where no floating point error can be tolerated.* Let R be the base ring of the computational model: i.e., all numerical input data (such as the coordinates of vertices, the coefficients of normals to $(n-1)$-faces etc.) come from R. Furthermore, we assume that $\mathbb{Z} \leq \mathsf{R} \leq \mathbb{R}$ in the partial order of rings. When we discuss the degrees of the polynomial predicates evaluated by the algorithm, we consider them as polynomials with integer coefficients in the input parameters. In this context the phrase *arithmetic operation* stands for any ring-theoretic operation $(+, -, \times)$.

Case 1: *the dimension n is fixed.* In this case all linear-algebraic computations can be done via determinants. Using determinants has an advantage of keeping the degrees of evaluated polynomial predicates at bay. Moreover, since in our algorithm the largest determinants are 3×3, the highest degree of evaluated predicates is 3. Thus, the arithmetic complexity of the algorithm is $O(\mathsf{f}_{n-3\,n-2})$ and the algorithm evaluates at most $O(\mathsf{f}_{n-3\,n-2})$ polynomial predicates of degree 3.

Case 2: *the dimension n is not fixed.* If n is not too large, the linear-algebraic computations can still be done via direct determinant evaluations. In each computation we are dealing with at most three n-vectors, which means that we may have to evaluate $\binom{n}{n-3}$ 3×3 determinants to find a minor of maximal rank. Thus, the total arithmetic complexity of the algorithm is $O(n^3\, \mathsf{f}_{n-3\,n-2})$. The case of large values of n is considered in the next paragraph.

7.1 Exact Computations over $\mathbb{Z}$.

In here we consider the case of exact computation. The dimension n is not fixed and $\mathsf{R} = \mathbb{Z}$. We are now interested in the bit complexity of our algorithm, e.g., in the multitape Turing machine model. Note that the same techniques can be used for sign computations in our algorithm. Since each sign computation involves no more than 6 vectors, the bit complexity of each sign computation using Yap's (2002) ramification of the Bachem-Kannan algorithm is $O(n\mathsf{M}_b(L))$, where $\mathsf{M}_b(x)$ is the bit-complexity of multiplication of two integers of binary sizes not exceeding x and L is a bound on the binary size of the coefficients of the vectors (see (Yap, 2002) for details). Then the total complexity of the algorithm is $O(n\mathsf{f}_{n-3\,n-2}\mathsf{M}_b(L))$. Devillers et al. (1998) have shown that any convexity checker, *whose work does not depend on the nature of* R, has to evaluate at least one polynomial of degree n – however, this lower bound is mandatory only for those checkers that work the same way for any R.

Theorem 11 *Let $r : \mathscr{M} \to \mathbb{R}^n$ be a cellwise-flat PL-realization of a manifold $(\mathscr{M}, \mathcal{P})$ of dimension $n-1$. Suppose the* input is in the standard form *and all normals have integer coordinates of binary size not exceeding L. There exists a polynomial time algorithm for checking convexity of $r : \mathscr{M} \to \mathbb{R}^n$ with (multitape Turing machine) complexity of $O(n\mathsf{f}_{n-3\,n-2}\mathsf{M}_b(L))$.*

Now, let us consider the situation where the input is given in the traditional form, i.e. as the poset $\mathcal{P}[0, n-3, n-2, n-1]$, equipped with the coordinates of the vertices. If we have no restrictions on the combinatorics and geometry of the geometric realization of cells of $(\mathcal{P}, \mathscr{M})$, then it is very difficult, or even impossible, to construct Euclidean inner normals from given data. Let us assume that the partition $\mathcal{P}$ is simplicial. In order to do sign computations, we need first write down *Euclidean* inner normals for all corner-facet and corner-ridge incidences. For each such incidence we have to deal with roughly n vectors of length n. Computing a Euclidean normal is then reduced to a fidning a non-zero solution for a homogeneous system $M\boldsymbol{x} = 0$ where M is at most n by n matrix. We can use Yap's (2002) version of the Bachem-Kannan algorithm to compute (upper triangular) Hermit's Normal Form for the system $M\boldsymbol{x} = 0$. Then a non-zero solution vector of at most polynomial size can be found in polynomial time by using standard techniques of linear algebra: we just work our way from the bottom of the normalized matrix up until all x_i's are found. Alternatively, one can use a polynomial algorithm in Yap (2002: Sec. 10.8-10.9), based on repeated application of Bachem-Kannan algorithm, to further reduce the system to Smith Normal Form and then find a solution. Furthermore, to reduce the complexity, we can deal with each corner C in the following way. If $\boldsymbol{v}_0, ..., \boldsymbol{v}_n$ are the vertices of $r(C)$, then we first find Hermit's normal form for the matrix $[\boldsymbol{v}_1 - \boldsymbol{v}_0, ..., \boldsymbol{v}_n - \boldsymbol{v}_0]$ and then, for each P, where P is a ridge or a facet incident to C, compute an integral Euclidean normal vector to $r(P)$ at $r(C)$. Then the complexity of all linear-algebraic computations for $\mathrm{Star}\, C$ is dominated by the complexity of finding Hermit's normal form for the matrix $[\boldsymbol{v}_1 - \boldsymbol{v}_0, ..., \boldsymbol{v}_n - \boldsymbol{v}_0]$, which is $O(n^3 \mathsf{M}_b(L))$ (Yap, 2002). Thus , the total complexity is $O(n^3 \mathsf{f}_{n-3\,n-2} \mathsf{M}_b(L))$.

If $n = 3$, then the corners are the vertices and the required normals are easy to produce. Now, suppose $n > 3$. How large the coefficients of the integral Euclidean normals, discussed above, can be? It is obviously possible to produce each such Euclidean normal as a vector whose coordinates are polynomials of degree at most $n-3$ in the coordinates of the vertices. Furthermore, Siegel (see e.g. Yap, 2002) proved that a homogeneous system of k linear equations with n variables over $\mathbb{Z}$ has a non-zero solution where each component is bounded in absolute value by $1 + (nA)^{\frac{k}{n-k}}$ (for us $k = n-3$) where A is the largest of the absolute values of the coefficients. Siegel also showed this bound could not be improved. When n is small enough, a vector satisfying Siegel's bound can be found by classical methods of lattice reduction (no efficient methods for finding such a vector are known for large n). If normals satsifying Siegel's bound are used in the algorithmic precedures given above, then the largest integers that may appear in sign computations via determinants are of the order $\lambda(nA_0)^{n-3}$ – where $\lambda \gtrapprox 1$ is a constant slightly larger than 1, and A_0 is twice the largest of the absolute values of the vertex coordinates.

Theorem 12 *Let $r : \mathscr{M} \longrightarrow \mathbb{R}^n$ be a cellwise-flat PL-realization of a* simplicial *manifold $(\mathscr{M}, \mathcal{P})$ of dimension $n-1$. Suppose the input consists of the poset $\mathcal{P}[0, n-3, n-2, n-1]$ equipped with the coordinates of the vertices, and that for each vertex $\boldsymbol{v} = r(v)$ of $r(\mathscr{M})$ we have $\boldsymbol{v} \in \mathbb{Z}^n$ and $|\boldsymbol{v}| \le 2^L$. There exists a polynomial time algorithm for checking convexity of $r : \mathscr{M} \longrightarrow \mathbb{R}^n$ with (multitape Turing machine) complexity of $O(n^3 \mathsf{f}_{n-3\,n-2} \mathsf{M}_b(L))$.*

The input requirements in the above theorem can be relaxed. If we know only $\mathcal{P}' = \mathcal{P}[0, n-3, n-2]$ (or $\mathcal{P}' = \mathcal{P}[0, n-3, n-1]$) together with the circular order of facets (or ridges) at all ridges, then $\mathcal{P}[0, n-3, n-2, n-1]$ can be computed at no extra cost.

7.2 Surfaces in $\mathbb{R}^3$

The algorithm runs in linear time in the number of vertices when $\mathscr{M}$ is spherical. However, a sequence of non-spherical PL-manifolds of dimension 2 can have the edge number growing quadratically in f_0. Thus, it is desirable to check the topological type of the input by just counting 1-cells (edges) in $\mathcal{P}[0, 1]$: once their number exceeds $c\mathsf{f}_0$, where c is some constant which is easy to calculate, we stop and declare the input non-convex. This check helps preserve the $O(\mathsf{f}_0)$ running time bound for PL-surfaces in $\mathbb{R}^3$. One may wonder if such a check is necessary, as it seems very likely our algorithm will quickly encounter a non-convex vertex, if the input surface is homeomorphic to a sphere-with-handles or sphere-with-Möbuis-strips. Surprisingly, Betke & Gritzmann (1984), proved that any orientable non-spherical connected closed 2-manifold can be PL-embedded into $\mathbb{R}^3$ so that it has exactly 5 non-convex vertices but no fewer! The problem of determining the minimal possible number of non-convex vertices in a PL-immersion of a non-orientable closed 2-manifold is open.

In the case of $\mathbb{R}^3$ the requirements on the combinatorial part of the input can be somewhat relaxed: in what follows we show it is sufficient to know only $\mathcal{P}[0,1]$, which is the 1-skeleton graph of $(\mathscr{M},\mathcal{P})$. First, the planarity of this graph can be checked in $O(\mathsf{f}_0)$ time (Lempel et al, 1967). For a planar graph we can also determine the faces in linear time – i.e., in $O(\mathsf{f}_0)$ time we can create the face-nodes, where each face-node is double-linked to its edge-nodes. Once we know the faces in terms of their edges, we can double link each face-node to the vertices-nodes of all of its edges. Because of the sphericity of $(\mathscr{M},\mathcal{P})$ the latter task takes $O(\mathsf{f}_0)$ time. Thus, the adjacency list representing $\mathcal{P}[0,1,2]$ can be constructed from the adjacency list representing $\mathcal{P}[0,1]$ in $O(\mathsf{f}_0)$ time.

The case of $\mathbb{R}^3$ is a rather special one. First, $n = 3$ is the smallest dimension for which the techniques of this paper apply. Second, even in the case of $\mathbb{R}^n$ the convexity test for each corner is reduced to testing convexity of a section of the star of this corner, which is essentially equivalent to testing convexity of a cone in $\mathbb{R}^3$. Another important consideration is that in application a 2-surface in $\mathbb{R}^3$ is normally specified by its combinatorics and the coordinates of the vertices or equations for the facets: it is therefore important to specify how our algorithm can be applied when the imput is given in the traditional form. Namely, suppose we are given $\mathcal{P}[0,1,2]$ equipped with the coordinates of the vertices $\boldsymbol{v}_1 = r(v_1),\ldots,\boldsymbol{v}_{\mathsf{f}_0} = r(v_{\mathsf{f}_0})$. The corner-ridge normals are then just vectors $\boldsymbol{v}_i - \boldsymbol{v}_j$. The question remains how to find corner-facet normals, i.e. vectors pointing from the vertices of the facets into the interiors of the facets. This is easy if it is known that the facets are convex. Otherwise we have the following algorithmic problem. Let C_k be the k-cycle graph. Consider a rectilinear embedding r of C_n in an affine plane $A \subset \mathbb{R}^3$ – the pair (C_k, r) defines a 2-dimensional polytope $P(C_k, r)$ whose boundary is $r(C_k)$ (here C_k is regarded as PL-manifold). Let v be a vertex of C_k. The problem is to find a non-zero vector $\boldsymbol{n} \in \overrightarrow{A}$ such that $r(v) + \varepsilon\boldsymbol{n}$ lies in the interior of $P(C_k, r)$. This problem can be solved in time $O(k)$; solving this problem for all facets will require $O(\mathsf{f}_0)$ ring-arithemtic operations. Thus, there is no difference in time-complexity betwen the standard and traditional forms of the input for $n = 3$. We will now restate the observations made in this Section in the following theorem.

Theorem 13 *Let $r : \mathscr{M} \to \mathbb{R}^3$ be a cellwise-flat PL-realization of a 2-manifold $(\mathscr{M},\mathcal{P})$. Suppose we are given the 1-skeleton of $(\mathscr{M},\mathcal{P})$ equipped with the coordinates of the vertices. Let L be the upper bound on the bit sizes of the coordinates of the vertices. There exists an algorithm for checking convexity of $r : \mathscr{M} \to \mathbb{R}^n$ with (multitape Turing machine) complexity of $O(\mathsf{f}_0\mathsf{M}_b(L))$.*

8 Conclusions

This paper describes a local approach to convexity verification of PL-hypersurfaces. The main theoretical result of the paper is a characterization of global convexity of a PL-hypersurface in $\mathbb{R}^n$ in terms of the local convexity properties of the surface at its $(n-3)$-faces. Building on this approach we give a polynomial-time convexity checking algorithm that can be applied for any closed hypersurface. The approach presented in this paper can be generalized to piecewise-polynomial surfaces of small degree.

References

A. D. Alexandrov: *The intrinsic geometry of convex surfaces*, OGIZ, (Moscow and Leningrad, 1948) [Russian] (reviewed by Busemann in *Math Reviews*, October 1949, **10**, No. 9, pp. 619–620).

U. Betke and P. Gritzmann, Polyedrische 2-Mannigfaltigkeiten mit wenigen nicht-konvexen Ecken [Polyhedral 2-manifolds with few nonconvex vertices], *Monatsh. Math.* (1984) **97**, No. 1, 1–21.

M. Blum and S. Kannan, Programs that check their work, In: *Proceedings of the Twenty First Annual ACM Symposium on Theory of Computing*, (1989), 86–97.

H. Busemann: *Convex Surfaces,* Interscience Publishers, (New York, 1958).

O. Devillers, G. Liotta, F. Preparata, and R. Tamassia, Checking the convexity of polytopes and the planarity of subdivisions, *Computational Geometry*, (1998) **11**, 187–208.

J. Dugundji: *Topology*, Prentice-Hall, (Englewood Cliffs, New Jersey, 1965).

L. Jonker, R. Norman, On Locally Convex Manifolds, *Canad. J. Math.* (1973) **25**, 531–538.

M. Joswig, G. Ziegler, Convex Hulls, Oracles, and Homology, *J. Symbolic Computation* (special issue for ICMS 2002) (2004), **38**, 1247–1259.

J. Van Heijenoort, On Locally Convex Manifolds, *Communications on Pure and Applied Mathematics*, (1952) vol. **V**, 223–242.

A.M. Kostrikin, Yu.I. Manin: *Linear Algebra and Geometry,* Gordon and Breach Science Pub., (1989)

A. Kuzminykh, *Convex bodies with paradoxical topological properties in Hyperbolic space.* 2005 Summer Conference on Topology and its Applications July 10-14, (2005), Denison University, Granville, Ohio.

A. Lempel, S. Even, and I. Cederbaum, An algorithm for planarity testing of graphs. In: P. Rosenstiehl, editor, *Theory of Graphs, International Symposium,* (1967) Rome, pp. 215–232.

K. Mehlhorn and S. Näher: LEDA: A Platform for Geometric and Combinatorial Computing, Cambridge Univ. Press. (2000)

K. Mehlhorn, S. Näher, M. Seel, R. Seidel, T. Schilz, S. Schirra, U. Stefan, and C. Uhrig, Checking Geometric Properties or Verification of Geometric Structures, In: *Proc. of 12th Annual ACM Sympos. Comput. Geom., Computational Geom.* (1996), 159–165.

K. Mehlhorn, S. Näher, M. Seel, R. Seidel, T. Schilz, S. Schirra, U. Stefan, and C. Uhrig, Checking Geometric Properties or Verification of Geometric Structures, *Computational Geometry*, (1999) **12**, No. 1-2, 85–103.

R. T. Rockafellar: *Convex Analysis*, Princeton Univ. Press, (Princeton, New Jersey, 1990).

H. Seifert and W. Threlfall: *A Textbook of Topology,* Academic Press (New York, N.Y., 1980) (translated from *Lehrbuch der Topologie*, Leipzig, 1934, by M. A. Goldman).

G. Ziegler, Face numbers of 4-polytopes and 3-spheres. In: *Proceedings of the International Congress of Mathematicians, Vol. III, Beijing, 2002,* 625–634, Higher Ed. Press, (Beijing, 2002).

C. Yap: *Fundamental Problems in Algorithmic Algebra,* Oxford Univ. Press, (2000).